

# Discrete-query quantum algorithm for NAND trees

Andrew M. Childs\*    Richard Cleve†‡    Stephen P. Jordan§    David Yeung†

16 February 2007

## Abstract

Recently, Farhi, Goldstone, and Gutmann gave a quantum algorithm for evaluating NAND trees that runs in time  $O(\sqrt{N \log N})$  in the Hamiltonian query model. In this note, we point out that their algorithm can be converted into an algorithm using  $O(N^{1/2+\epsilon})$  queries in the conventional quantum query model, for any fixed  $\epsilon > 0$ .

A NAND tree of depth  $n$  is a balanced binary tree whose internal vertices represent NAND gates. Placing bits  $x_1, \dots, x_{2^n}$  at the leaves, the root of the NAND tree evaluates to the function  $f_n(x_1, \dots, x_{2^n})$ , where  $f_n : \{0, 1\}^{2^n} \rightarrow \{0, 1\}$  is defined recursively as follows. For  $n = 0$ ,  $f_0(x) = x$ , and for  $n > 0$ ,

$$f_n(x_1, \dots, x_{2^n}) = \neg(f_{n-1}(x_1, \dots, x_{2^{n-1}}) \wedge f_{n-1}(x_{2^{n-1}+1}, \dots, x_{2^n})). \quad (1)$$

The goal of the NAND tree problem is to evaluate  $f_n(x_1, \dots, x_{2^n})$ , making as few queries to the bits  $x_1, \dots, x_{2^n}$  as possible. The optimal classical randomized algorithm for this problem makes  $\Theta(N^{0.753})$  queries, where  $N = 2^n$  [7, 8, 9]. Until now, no better quantum algorithm was known, whereas the best known quantum lower bound is only  $\Omega(\sqrt{N})$  [1]. Here we show that for any fixed  $\epsilon > 0$ , the quantum query complexity of evaluating NAND trees is  $O(N^{1/2+\epsilon})$ .

Very recently, Farhi, Goldstone, and Gutmann [4] proposed a quantum algorithm that evaluates NAND trees in time  $O(\sqrt{N \log N})$ , albeit in the unconventional Hamiltonian oracle model [5, 6]. In their version of this model, we are given access to a Hamiltonian  $H_O$  acting on  $n + 1$  qubits as

$$H_O|b, k\rangle = -x_k|-b, k\rangle \quad (2)$$

for all  $b \in \{0, 1\}$  and  $k \in \{0, 1\}^n$ , and the goal is to perform the computation using evolution according to  $H_O + H_D(t)$  for as short a time as possible, where  $H_D(t)$  is an arbitrary driving Hamiltonian (that is possibly time-dependent and may act on an extended Hilbert space).

In the conventional quantum query model, the input is accessible via unitary operations of the form

$$U_O|k, a\rangle = |k, a \oplus x_k\rangle. \quad (3)$$

Two queries of  $U_O$  can be used to implement evolution according to  $H_O$  for an arbitrary time  $t$ , which can be seen as follows. The procedure acts on states of the form  $|b, k, a\rangle$  (where the last register is an ancilla qubit), and consists of the following steps. First, apply  $U_O$  to the second and third registers. Then apply a controlled- $R(t)$  gate with the first register as the target and the third register as the control, where

$$R(t) = \begin{pmatrix} \cos t & i \sin t \\ i \sin t & \cos t \end{pmatrix}. \quad (4)$$

Finally, apply  $U_O$  to the second and third registers again. With the ancilla qubit initially in the  $|0\rangle$  state, the net effect of this procedure is the mapping  $|b, k, 0\rangle \mapsto \cos(x_k t)|b, k, 0\rangle + i \sin(x_k t)|-b, k, 0\rangle$ , which corresponds to evolution by  $H_O$  for time  $t$  (that is, the unitary operation  $e^{-iH_O t}$ ).

\*Institute for Quantum Information, California Institute of Technology

†David R. Cheriton School of Computer Science and Institute for Quantum Computing, University of Waterloo

‡Perimeter Institute for Theoretical Physics

§Center for Theoretical Physics, Massachusetts Institute of Technology

This simulation of  $H_O$  does not imply that any fast algorithm in the Hamiltonian oracle model can be turned into an algorithm with small query complexity in the conventional quantum query model. Accurate simulation of the evolution according to  $H_O + H_D(t)$  apparently requires many interleaved evolutions of  $H_O$  and  $H_D(t)$  each for a small time, yet each of which requires two unitary queries to simulate. Nevertheless, it turns out that a Hamiltonian of the kind used in [4] *can* be simulated in the conventional quantum query model with only small overhead.

In the algorithm of [4],  $H_D(t)$  is time-independent, so the evolution for time  $t$  is given by  $e^{-i(H_O + H_D)t}$ . Such evolution according to a sum of time-independent Hamiltonians can be simulated using a high-order approximation of the exponential of a sum in terms of a product of exponentials of the individual terms. As noted in [2, 3], by using a  $p^{\text{th}}$  order approximation, the simulation can be performed in  $O((ht)^{1+1/2p})$  steps, where  $h = \max\{\|H_O\|, \|H_D\|\}$ , and where the constant depends on  $p$  and on the desired simulation error. We have  $\|H_O\| \leq 1$  and  $\|H_D\| \leq 3$ , so  $h \leq 3$ . Choosing  $p$  to be an arbitrarily large constant, we obtain a simulation using  $O(t^{1+\delta})$  steps, where  $\delta = 1/2p$  is arbitrarily small. Since the algorithm of [4] applies  $H$  for time  $t = O(\sqrt{N \log N})$ , it follows that the corresponding time evolution can be simulated using  $O(N^{1/2+\epsilon})$  queries to the original oracle, where  $\epsilon > 0$  may be arbitrarily small.

The result can also be deduced by noting that, given query access to the inputs via  $U_O$  (Eq. 3), one can easily simulate an oracle for the matrix elements of the underlying Hamiltonian  $H_O + H_D$  used in [4], and then apply results in [2, 3] for simulating sparse Hamiltonians.

## Acknowledgments

AMC received support from the U.S. NSF under grant no. PHY-0456720, and from the U.S. ARO under grant no. W911NF-05-1-0294. RC received support from Canada's CIAR, MITACS, NSERC, and the U.S. ARO/DTO. SPJ received support from the U.S. ARO/DTO's QuaCGR program. DY received support from the U.S. ARO/DTO.

## References

- [1] H. Barnum and M. Saks, *A lower bound on the quantum query complexity of read-once functions*, Journal of Computer and System Sciences **69**(2): 244–258 (2004), arXiv:quant-ph/0201007.
- [2] D. W. Berry, G. Ahokas, R. Cleve, and B. C. Sanders, *Efficient quantum algorithms for simulating sparse Hamiltonians*, Communications in Mathematical Physics **270**(2): 359–371 (2007), arXiv:quant-ph/0508139.
- [3] A. M. Childs, *Quantum information processing in continuous time*, Ph.D. thesis, Massachusetts Institute of Technology (2004).
- [4] E. Farhi, J. Goldstone, and S. Gutmann, *A quantum algorithm for the Hamiltonian NAND tree*, arXiv:quant-ph/0702144.
- [5] E. Farhi and S. Gutmann, *Analog analogue of a digital quantum computation*, Physical Review A **57**(4): 2403–2406 (1998), arXiv:quant-ph/9612026.
- [6] C. Mochon, *Hamiltonian oracles*, arXiv:quant-ph/060203.
- [7] M. E. Saks and A. Wigderson, *Probabilistic boolean decision trees and the complexity of evaluating game trees*, Proc. 27th IEEE Symposium on Foundations of Computer Science, pp. 29–38 (1986).
- [8] M. Santha, *Probabilistic boolean decision trees and the complexity of evaluating game trees*, Proc. 6th IEEE Structure in Complexity Theory Conference, pp. 180–187 (1991).
- [9] M. Snir, *Lower bounds for probabilistic linear decision trees*, Theoretical Computer Science **38**(1): 69–82 (1985).